Refine Search

Search Results -

Terms	Documents
L21 and L20 and L18 and L17	2

Database: US Pre-Grant Publication Full-Text Database; US Patents Full-Text Database; US OCR Full-Text Database; EPO Abstracts Database; JPO Abstracts Database; Derwent World Patents Index; IBM Technical Disclosure Bulletins

Search: L22

Refine Search | Recall Text | Clear | Interrupt

Search History

DATE: Saturday, September 04, 2004 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=ADJ			
L22	L21 and l20 and l18 and l17	2	L22
L21	L19 with l14	291	L21
L20	L19 with l16	153	L20
L19	clock adj3 buffer	1617	L19
L18	L14 near6 (chang$3 or state or differen$4 or modifi$6 or alter$6)	97923	L18
L17	L16 with l15 with l14	85	L17
L16	internal near4 (clock or pulse)	7889	L16
L15	input near4 (buffer or register)	53737	L15
L14	(input or receiv$3) near4 signal	794816	L14
DB=PGPB,USPT; PLUR=YES; OP=ADJ			
L13	L12 and (data near4 buffer)	76	L13
L12	L11 and (address near4 buffer)	82	L12
L11	L10 and (command near4 buffer)	84	L11
L10	L9 and l4	228	L10
L9	L8 and l7 and l6	397	L9
L8	L5 with l3	1118	L8
L7	L5 with l1	1926	L7
L6	l1 near6 (chang$3 or state or differen$4 or modifi$6 or alter$6)	168198	L6

L5	clock adj3 buffer	6991	L5
L4	L3 with l2 with l1	703	L4
L3	internal near4 (clock or pulse)	26659	L3
L2	input near4 (buffer or register)	103583	L2
L1	(input or receiv$3) near4 signal	689122	L1

END OF SEARCH HISTORY

Refine Search

Search Results -

Terms	Documents
L12 and L10 and L9	1

Database: US Pre-Grant Publication Full-Text Database; US Patents Full-Text Database; US OCR Full-Text Database; EPO Abstracts Database; JPO Abstracts Database; Derwent World Patents Index; IBM Technical Disclosure Bulletins

Search: L13

Refine Search | Recall Text | Clear | Interrupt

Search History

DATE: Friday, September 03, 2004 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=ADJ			
L13	L12 and l10 and l9	1	L13
L12	L11 with ((input adj2 signal) near6 (chang$4 or alter$6 or modifi$8 or state))	26	L12
L11	(input adj2 signal) with (internal near4 (clock or pulse))	326	L11
L10	((input or read or I/O) near4 (buffer or register)) with (internal near4 clock)	191	L10
L9	L8 near8 (internal near3 clock)	1834	L9
L8	clock near4 (buffer or register or generat$4)	70503	L8
DB=USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=ADJ			
L7	clock near4 (buffer or register or generat$4)	161166	L7
DB=PGPB,USPT; PLUR=YES; OP=ADJ			
L6	L5 and l3 and l2	26	L6
L5	L4 with ((input adj2 signal) near6 (chang$4 or alter$6 or modifi$8 or state))	90	L5
L4	(input adj2 signal) with (internal near4 (clock or pulse))	1375	L4
L3	((input or read or I/O) near4 (buffer or register)) with (internal near4 clock)	1427	L3
L2	L1 near8 (internal near3 clock)	6381	L2
L1	clock near4 (buffer or register or generat$4)	107853	L1

END OF SEARCH HISTORY

Welcome to IEEE Xplore®
- Home
- What Can I Access?
- Log-out

Tables of Contents
- Journals & Magazines
- Conference Proceedings
- Standards

Search
- By Author
- Basic
- Advanced

Member Services
- Join IEEE
- Establish IEEE Web Account
- Access the IEEE Member Digital Library

IEEE Enterprise
- Access the IEEE Enterprise File Cabinet

Your search matched **0** of **1067317** documents.
A maximum of **500** results are displayed, **15** to a page, sorted by **Relevance** in **Descending** order.

Refine This Search:
You may refine your search by editing the current search expression or entering a new one in the text box.

((input signal) <near/4> chang$3) <sentence> clock buf Search

☐ Check to search within this result set

Results Key:
JNL = Journal or Magazine **CNF** = Conference **STD** = Standard

---

Results:
No documents matched your query.

Welcome to IEEE Xplore®
- Home
- What Can I Access?
- Log-out

Tables of Contents
- Journals & Magazines
- Conference Proceedings
- Standards

Search
- By Author
- Basic
- Advanced

Member Services
- Join IEEE
- Establish IEEE Web Account
- Access the IEEE Member Digital Library

IEEE Enterprise
- Access the IEEE Enterprise File Cabinet

Your search matched **0** of **1067317** documents.
A maximum of **500** results are displayed, **15** to a page, sorted by **Relevance** in **Descending** order.

Refine This Search:
You may refine your search by editing the current search expression or entering a new one in the text box.

((input signal) <near/4> chang$) <sentence> clock buff Search

☐ Check to search within this result set

Results Key:
JNL = Journal or Magazine **CNF** = Conference **STD** = Standard

Results:
No documents matched your query.

IEEE Xplore® RELEASE 1.8

Welcome
United States Patent and Trademark Office

» Advanced Searc

Help FAQ Terms IEEE Peer Review Quick Links

Welcome to IEEE Xplore®
- Home
- What Can I Access?
- Log-out

Tables of Contents
- Journals & Magazines
- Conference Proceedings
- Standards

Search
- By Author
- Basic
- Advanced

Member Services
- Join IEEE
- Establish IEEE Web Account
- Access the IEEE Member Digital Library

IEEE Enterprise
- Access the IEEE Enterprise File Cabinet

Try our New Full-text Search Prototype GO Help

1) Enter a single keyword, phrase, or Boolean expression.
Example: acoustic imaging (means the phrase acoustic imaging plus any stem variations)
2) Limit your search by using search operators and field codes, if desired.
Example: optical <and> (fiber <or> fibre) <in> ti
3) Limit the results by selecting Search Options.
4) Click Search. See Search Examples

```
((input signal) <near/4>
chang$) <sentence> clock
buffer <paragraph> (generat$
<near/4> (internal clock))
```

Start Search Clear

Note: This function returns plural and suffixed forms of the keyword(s).

Search operators: <and> <or> <not> <in> More

Field codes: au (author), ti (title), ab (abstract), jn (publication name), de (index term) More

Search Options:

Select publication types:

- [x] IEEE Journals
- [x] IEE Journals
- [x] IEEE Conference proceedings
- [x] IEE Conference proceedings
- [x] IEEE Standards

Select years to search:

From year: All to Present

Organize search results by:

Sort by: Relevance
In: Descending order

List 15 Results per page

Terms used
input signal near/4 chang sentence clock buffer paragraph generat near/4 internal clock

Found **12,183** of **141,680**

Sort results by relevance

Display results expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown

Relevance scale

1 An Unclever Time-Sharing System

Caxton C. Foster

January 1971 **ACM Computing Surveys (CSUR)**, Volume 3 Issue 1

Full text available: pdf(1.85 MB) Additional Information: full citation, abstract, references, citings, index terms

This paper describes the internal structure of a time-sharing system in some detail. This system is dedicated to providing remote access, and has a simple file structure. It is intended for use in a university type environment where there are many short jobs that will profit from one- or two-second turnaround. Despite its simplicity, this system can serve as a useful introduction to the problems encountered by the designers of any time-sharing system. Included are a discussion of the comman ...

2 Interactive Editing Systems: Part II

Norman Meyrowitz, Andries van Dam

September 1982 **ACM Computing Surveys (CSUR)**, Volume 14 Issue 3

Full text available: pdf(9.17 MB) Additional Information: full citation, references, citings, index terms

3 Fast detection of communication patterns in distributed executions

Thomas Kunz, Michiel F. H. Seuren

November 1997 **Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research**

Full text available: pdf(4.21 MB) Additional Information: full citation, abstract, references, index terms

Understanding distributed applications is a tedious and difficult task. Visualizations based on process-time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

4 Contention resolution with constant expected delay

Leslie Ann Goldberg, Philip D. Mackenzie, Mike Paterson, Aravind Srinivasan

November 2000 **Journal of the ACM (JACM)**, Volume 47 Issue 6

Full text available: pdf(388.69 KB) Additional Information: full citation, abstract, references, citings, index terms

We study contention resolution in a multiple-access channel such as the Ethernet channel. In the model that we consider, n users generate messages for the channel according to a probability distribution. Raghavan and Upfal have given a protocol in which the expected delay (time to get serviced) of every message is O(log n) when messages are generated

according to a Bernoulli distribution with generation rate up to about 1/10. Our main resul ...

Keywords: Markov chains, contention resolution, ethernet, multiple-access channel

5 Abstract interaction tools: a language for user interface management systems

Jan Van Den Bos

April 1988 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 10 Issue 2

Full text available: pdf(2.45 MB) Additional Information: full citation, abstract, references, citings, index terms, review

A language model is presented for the specification of User Interface Management Systems. The model, called the Abstract Interaction Tool (AIT) model, offers a tree-like hierarchy of interaction objects. Each object represents a subtree and can be considered as an abstract input device containing a syntax-like specification of the required input pattern. The hierarchy of specifications amounts to a system of syntactical productions with multiple control. Terminal nodes of the AIT tree repre ...

6 Basic elements of COBOL 61

Jean E. Sammet

May 1962 **Communications of the ACM**, Volume 5 Issue 5

Full text available: pdf(1.70 MB) Additional Information: full citation, references, citings

7 SpeechSkimmer: a system for interactively skimming recorded speech

Barry Arons

March 1997 **ACM Transactions on Computer-Human Interaction (TOCHI)**, Volume 4 Issue 1

Full text available: pdf(1.03 MB) Additional Information: full citation, abstract, references, citings, index terms, review

Listening to a speech recording is much more difficult than visually scanning a document because of the transient and temporal nature of audio. Audio recordings capture the richness of speech, yet it is difficult to directly browse the stored information. This article describes techniques for structuring, filtering, and presenting recorded speech, allowing a user to navigate and interactively find information in the audio domain. This article describes the SpeechSkimmer system for interacti ...

Keywords: audio browsing, interactive listening, nonspeech audio, speech as data, speech skimming, speech user interfaces, time compression

8 The early history of COBOL

Jean E. Sammet

January 1978 **ACM SIGPLAN Notices , The first ACM SIGPLAN conference on History of programming languages**, Volume 13 Issue 8

Full text available: pdf(3.10 MB) Additional Information: full citation, abstract, references, citings, index terms

This paper discusses the early history of COBOL, starting with the May 1959 meeting in the Pentagon which established the Short Range Committee which defined the initial version of COBOL, and continuing through the creation of COBOL 61. The paper gives a detailed description of the committee activities leading to the publication of the first official version, namely COBOL 60. The major inputs to COBOL are discussed, and there is also a description of how and why some of the technical decisi ...

9 Development and application of NASA's first standard spacecraft computer

Charles E. Trevathan, Thomas D. Taylor, Raymond G. Hartenstein, Ann C. Merwarth, William N. Stewart

September 1984 **Communications of the ACM**, Volume 27 Issue 9

Full text available: pdf(1.26 MB) Additional Information: full citation, abstract, citings, index terms

To provide the autonomy needed by low, earth-orbiting satellites, NASA's first standard on-board processor requires changing only interfacing hardware from mission to mission.

Keywords: PASS, avionics system

10 A new string search hardware architecture for VLSI

K. Takahashi, H. Yamada, H. Nagai, K. Matsumi

June 1986 **ACM SIGARCH Computer Architecture News , Proceedings of the 13th annual international symposium on Computer architecture**, Volume 14 Issue 2

Full text available: pdf(683.39 KB) Additional Information: full citation, abstract, references, citings, index terms

This paper presents a new architecture for practical string search hardware design. This architecture is based on the finite state automaton design concept using a character control charge transfer model. The resultant hardware is a set of programmable sequential logic (PSL) circuits, each of which consists of a sequential logic and memory parts. The logic part is an array of logical gates, each of which is controlled by the read-out signal from the memory part, to connect the flip-flops. T ...

11 Architectures: A perspective on the future of massively parallel computing: fine-grain vs. coarse-grain parallel models comparison & contrast

Predrag T. Tosic

April 2004 **Proceedings of the first conference on computing frontiers on Computing frontiers**

Full text available: pdf(277.49 KB) Additional Information: full citation, abstract, references, index terms

Models, architectures and languages for *parallel computation* have been of utmost research interest in computer science and engineering for several decades. A great variety of parallel computation models has been proposed and studied, and different parallel and distributed architectures designed as some possible ways of harnessing parallelism and improving performance of the general purpose computers.*Massively parallel connectionist models* such as *artificial neural networks* (...

Keywords: cellular automata, distributed systems, massively parallel computing, multiprocessor computers, neural networks, parallel computation models

12 Abstract state machines capture parallel algorithms

Andreas Blass, Yuri Gurevich

October 2003 **ACM Transactions on Computational Logic (TOCL)**, Volume 4 Issue 4

Full text available: pdf(610.28 KB) Additional Information: full citation, abstract, references, index terms

We give an axiomatic description of parallel, synchronous algorithms. Our main result is that every such algorithm can be simulated, step for step, by an abstract state machine with a background that provides for multisets.

Keywords: ASM thesis, Parallel algorithm, abstract state machine, postulates for parallel computation

13 Concurrency, latency, or system overhead: which has the largest impact on uniprocessor DRAM-system performance?

Vinodh Cuppu, Bruce Jacob

May 2001 **ACM SIGARCH Computer Architecture News , Proceedings of the 28th annual international symposium on Computer architecture**, Volume 29 Issue 2

Full text available: pdf(904.17 KB) Publisher Site Additional Information: full citation, abstract, references, citings, index terms

Given a fixed CPU architecture and a fixed DRAM timing specification, there is still a large

design space for a DRAM system organization. Parameters include the number of memory channels, the bandwidth of each channel, burst sizes, queue sizes and organizations, turnaround overhead, memory-controller page protocol, algorithms for assigning request priorities and scheduling requests dynamically, etc. In this design space, we see a wide variation in application execution times: for example, ...

14 Multimode scan: Test per clock BIST for IP cores

Adit D. Singh, Markus Seuring, Michael Gössel, Egor S. Sogomonyan

October 2003 **ACM Transactions on Design Automation of Electronic Systems (TODAES)**, Volume 8 Issue 4

Full text available: pdf(168.82 KB) Additional Information: full citation, abstract, references, index terms

Built-in self-test (BIST) is an attractive design-for-test methodology for core-based SoC design because of the minimal need for test access when tests are generated and evaluated within the core itself. However, the scan based logic BIST approach being widely considered for this application suffers from two significant weaknesses: slow test-per-scan execution, and a limited capability for detecting realistic timing and delay faults, critical in deep submicron technologies. The new multimode sca ...

Keywords: BIST, SoC, digital testing, scan

15 Bibliography of recent publications on computer communication

Martha Steenstrup

January 1998 **ACM SIGCOMM Computer Communication Review**, Volume 28 Issue 1

Full text available: pdf(2.02 MB) Additional Information: full citation, abstract, index terms

The quantitative results presented in our SIGCOMM '97 paper [1] include numerous minor errors. These errors were caused by programming bugs that led to faulty analyses and simulations, and by inaccurate transcriptions during the preparation of the paper. Here we present corrected figures and tables, as well as corrections to values that appeared in the text of the original paper. The effect of correcting the errors is to reduce the differences between the results based on the proxy trace and tho ...

16 Complexity-effective superscalar processors

Subbarao Palacharla, Norman P. Jouppi, J. E. Smith

May 1997 **ACM SIGARCH Computer Architecture News , Proceedings of the 24th annual international symposium on Computer architecture**, Volume 25 Issue 2

Full text available: pdf(2.21 MB) Additional Information: full citation, abstract, references, citings, index terms

The performance tradeoff between hardware complexity and clock speed is studied. First, a generic superscalar pipeline is defined. Then the specific areas of register renaming, instruction window wakeup and selection logic, and operand bypassing are analyzed. Each is modeled and Spice simulated for feature sizes of 0.8µm, 0.35µm, and 0.18µm. Performance results and trends are expressed in terms of issue width and window size. Our analysis indicates that window wakeu ...

17 The design of system architectures for information retrieval

L. A. Hollaar, B. J. Hurley, D. J. Kuck, D. H. Lawrie, J. W.S. Liu, J. M. Milner, J. K. Morgan, J. R. Rinewalt, W. H. Stellhorn

October 1976 **Proceedings of the annual conference**

Full text available: pdf(638.08 KB) Additional Information: full citation, abstract, references, citings, index terms

A broad-based program for studying hardware, software and human factors affecting information retrieval system performance is described. Using EUREKA, a minicomputer-based retrieval program, studies of user adaptation to an on-line system, search strategy development and use of special features are in progress. Performance comparisons are made between machine-assisted users and those using conventional printed materials. Concurrently, special-purpose retrieval hardware is being designed, an ...

18 Flexible collaboration transparency: supporting worker independence in replicated application-sharing systems

James Begole, Mary Beth Rosson, Clifford A. Shaffer

June 1999 **ACM Transactions on Computer-Human Interaction (TOCHI)**, Volume 6 Issue 2

Full text available: pdf(312.22 KB) Additional Information: full citation, abstract, references, citings, index terms, review

This article presents a critique of conventional collaboration transparency systems, also called "application-sharing" systems, which provide the real-time shared use of legacy single-user applications. We find that conventional collaboration transparency systems are inefficient in their use of network resources and lack support for key groupware principles: concurrent work, relaxed WYSIWIS, and group awareness. Next, we present an alternative approach to implementing collaborat ...

Keywords: Flexible JAMM, Java, application sharing, collaboration transparency, computer-supported cooperative work, groupware, usability

19 Interconnect pipelining in a throughput-intensive FPGA architecture

Amit Singh, Arindam Mukherjee, Malgorzata Marek-Sadowska

February 2001 **Proceedings of the 2001 ACM/SIGDA ninth international symposium on Field programmable gate arrays**

Full text available: pdf(203.52 KB) Additional Information: full citation, abstract, references, citings, index terms

Wave-steering is a new design methodology that realizes high throughput circuits by embedding layout friendly synthesized structures in silicon. In the wave-steering design methodology, cirðcuits inherently utilize latches. Inside the synthesized structures they are used for signal skewing, and on the interconnects to guarðantee the correct arrival times at the inputs. Recently, we proposed a novel high-throughput FPGA architecture based on the wave-steering design principle to handle throu ...

20 On embedding a microarchitectural design language within Haskell

John Launchbury, Jeffrey R. Lewis, Byron Cook

September 1999 **ACM SIGPLAN Notices , Proceedings of the fourth ACM SIGPLAN international conference on Functional programming**, Volume 34 Issue 9

Full text available: pdf(1.26 MB) Additional Information: full citation, abstract, references, citings, index terms

Based on our experience with modelling and verifying microarchitectural designs within Haskell, this paper examines our use of Haskell as host for an embedded language. In particular, we highlight our use of Haskell's lazy lists, type classes, lazy state monad, and unsafe Perform I0, and point to several areas where Haskell could be improved in the future. We end with an example of a benefit gained by bringing the functional perspective to microarchitectural modelling.